UNITED STATES PATENT APPLICATION
FOR

METHOD AND APPARATUS For Verifying the Integrity of a Content-Addressable Memory Result

INVENTOR:

MARK A. ROSS

3344 MELENDY DRIVE
SAN CARLOS, CA 94070
A CITIZEN OF THE UNITED STATES

PREPARED BY:
THE LAW OFFICE OF KIRK D. WILLIAMS
1234 S. OGDEN ST.
DENVER, CO 80210
303-282-0151

EXPRESS MAIL CERTIFICATE OF MAILING

"Express Mail" mailing label number: EL759042835US
Date of Deposit: July 20, 2001
I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post Office to Addressee" service on the date indicated above and that this paper or fee has been addressed to BOX PATENT APPLICATION, ASST COMMISSIONER FOR PATENTS, WASHINGTON DC 20231.

Kirk D. Williams
(Typed or printed name of person mailing paper or fee)

(Signature of person mailing paper or fee)

July 20, 2001
(Date signed)

METHOD AND APPARATUS FOR VERIFYING THE INTEGRITY OF A CONTENT-ADDRESSABLE MEMORY RESULT

FIELD OF THE INVENTION

This invention especially relates to content-addressable memory devices, and communications and computer systems that employ content-addressable memories; and more particularly, the invention relates to verifying the integrity of a result (e.g., index or application result) produced by a content-addressable memory, associative memory, or other device.

BACKGROUND OF THE INVENTION

The communications industry is rapidly changing to adjust to emerging technologies and ever increasing customer demand. This customer demand for new applications and increased performance of existing applications is driving communications network and system providers to employ networks and systems having greater speed and capacity (e.g., greater bandwidth). In trying to achieve these goals, a common approach taken by many communications providers is to use packet switching technology. Increasingly, public and private communications networks are being built and expanded using various packet technologies, such as Internet Protocol (IP).

A network device, such as a switch or router, typically receives, processes, and forwards or discards a packet based on one or more criteria, including the type of protocol used by the packet, addresses of the packet (e.g., source, destination, group), and type or quality of service requested. Additionally, one or more security operations are typically performed on each packet. But before these operations can be performed, a packet classification operation must typically be performed on the packet.

Packet classification as required for access control lists (ACLs) and forwarding decisions is a demanding part of switch and router design. This packet classification of a received packet is increasingly becoming more difficult due to ever increasing packet rates and number of packet classifications. For example, ACLs require matching packets on a subset of fields of the packet flow label, with the semantics of a sequential search through the ACL rules. IP forwarding requires a longest prefix match.

One known approach uses binary and/or ternary content-addressable memories to perform packet classification. Ternary content-addressable memories allow the use of wildcards in performing their matching, and thus are more flexible than binary content-addressable memories.

However, content-addressable memories are made of storage elements that are subject to data errors. A corrupted bit or bits in an entry in a content-addressable memory ("CAM") can lead to an incorrect CAM result, and thus an error in routing, forwarding, quality of service or other characterization or application. Needed are mechanisms for verifying the integrity of a CAM result.

SUMMARY OF THE INVENTION

Systems and methods are disclosed verifying the integrity of a result produced by a content-addressable memory or other device. In one embodiment, an index is generated by a content-addressable memory based on an input value. A mask value and a data protection field are acquired based on the index. A comparison value is generated based on the mask value and the input value. The comparison value is compared to the data protection field. In one embodiment, a content-addressable memory index is received. A value field and a data protection field are extracted from the content-addressable memory index. A data protection function is performed on the value to generate a comparison result, and the comparison result is compared with the data protection field. In one embodiment, an index is generated by a content-addressable memory based on an input value. A comparison value is generated based on the index. A data protection field is acquired based on the index; and the comparison value is compared to the data protection field.

BRIEF DESCRIPTION OF THE DRAWINGS

The appended claims set forth the features of the invention with particularity. The invention, together with its advantages, may be best understood from the following detailed description taken in conjunction with the accompanying drawings of which:

FIG. 1 is a block diagram of one embodiment of data protected CAM;

FIGs. 2, 3A-C and 4 are block diagrams of embodiments for protecting the integrity of an index (also referred to herein as a result, and vice versa) generated by a content-addressable memory; and

FIGs. 5A-C are flow diagrams of exemplary process used in some of numerous embodiments for protecting the integrity of an index generated by a content-addressable memory.

DETAILED DESCRIPTION

Methods and apparatus are disclosed for verifying the integrity of an index or result produced by a content-addressable memory, associative memory, or other device. Embodiments described herein include various elements and limitations, with no one element or limitation contemplated as being a critical element or limitation. Each of the claims individually recite an aspect of the invention in its entirety. Moreover, some embodiments described may include, but are not limited to, *inter alia*, systems, networks, integrated circuit chips, embedded processors, ASICs, methods, and computer-readable medium containing instructions. The embodiments described hereinafter embody various aspects and configurations within the scope and spirit of the invention, with the figures illustrating exemplary and non-limiting configurations.

As used herein, the term "packet" refers to packets of all types, including, but not limited to, fixed length cells and variable length packets, each of which may or may not be divisible into smaller packets or cells. Moreover, these packets may contain one or more types of information, including, but not limited to, voice, data, video, and audio information. Furthermore, the term "system" is used generically herein to describe any number of components, elements, sub-systems, devices, packet switch elements, packet switches, routers, networks, computer and/or communication devices or mechanisms, or combinations of components thereof. The term "computer" is used generically herein to describe any number of computers, including, but not limited to personal computers, embedded processors and systems, control logic, ASICs, chips, workstations, mainframes, etc. The term "device" is used generically herein to describe any type of mechanism, including a computer or system or component thereof. The terms "task" and "process" are used generically herein to describe any type of running program, including, but not limited to a computer process, task, thread, executing application, operating system, user process, device driver, native code, machine or other language, etc., and can be interactive and/or non-interactive, executing locally and/or remotely, executing in foreground and/or background, executing in the user and/or operating system address

spaces, a routine of a library and/or standalone application, and is not limited to any particular memory partitioning technique. The steps and processing of signals and information illustrated in the figures are typically be performed in a different serial or parallel ordering and/or by different components in various embodiments in keeping within the scope and spirit of the invention. Moreover, the terms "network" and "communications mechanism" are used generically herein to describe one or more networks, communications mediums or communications systems, including, but not limited to the Internet, private or public telephone, cellular, wireless, satellite, cable, local area, metropolitan area and/or wide area networks, a cable, electrical connection, bus, etc., and internal communications mechanisms such as message passing, interprocess communications, shared memory, etc. The terms "first," "second," etc. are typically used herein to denote different units (e.g., a first element, a second element). The use of these terms herein does not necessarily connote an ordering such as one unit or event occurring or coming before the another, but rather provides a mechanism to distinguish between particular units. Moreover, the phrase "based on x" is used to indicate a minimum set of items x from which something is derived, wherein "x" is extensible and does not necessarily describe a complete list of items on which the operation is based. Additionally, the phrase "coupled to" is used to indicate some level of direct or indirect connection between two elements or devices, with the coupling device or devices modify or not modifying the coupled signal or communicated information.

Methods and apparatus are disclosed for verifying the integrity of an index or result produced by a content-addressable or associated memory or other device. A pre-computed data protection field is stored, either as part of a returned index of a content-addressable memory or in a separate storage. In one embodiment, a data protection operation is performed on all or part of the returned index and a comparison is made with a pre-computed data protection field. In one embodiment, a copy of the masks employed by a ternary content-addressable memory and a set of pre-computed data protection fields are stored. A particular mask and pre-computed data protection field are

selected based on the generated index. The original input value is then masked by the selected mask and provided to a data protection function. The result of this function is then compared to the selected pre-computed data protection field.

Embodiments of the invention include, but are not limited to a single physical device having incorporated therein the data lookup and protection mechanisms as well as multiple components or physical devices. FIG. 1 illustrates one embodiment of a data protected content-addressable memory (CAM) 100, wherein the data protection functionality of the invention is included in a single device. FIG. 2 illustrates one embodiment which uses an adjunct data protection mechanism 210 (e.g., circuitry, processors, logic, etc.) for performing data protection functionality.

As shown in FIG. 1, data protected CAM 100 receives an input word 101 (e.g., any value or number of bits), and generates an index or result (e.g., ACL, forwarding, or other indication) 107, along with a valid flag 106, a hit flag 108 and error flag 109. Hit flag 108 is used to indicate whether a match was identified, and error flag 109 is used to indicate whether the produced index or result is valid. Valid flag 106 is used to identify when signals on index or result 107, hit flag 108, and error flag 109 indicate an actual result.

As shown in FIG. 2, CAM 202 receives an input word 201 and generates an index 207 (or result) which is provided to data protection mechanism 210. In some embodiments, input word 201 is also provided to data protection mechanism 210, while in some embodiments it is not. Data protection mechanism 210 controls the values of hit flag 208 which is used to indicate whether a match was identified, and of error flag 209 which is used to indicate whether the produced index or result is valid. Valid flag 206 is used to identify when signals on index 207, hit flag 208, and error flag 209 indicate an actual result.

FIG. 3A illustrates one embodiment of a system for verifying the integrity of an index 307 generated by binary CAM 302 in response to received input word 301. In other embodiments, index 307 corresponds to a data result rather than a CAM index, and binary

CAM may be another type of CAM, associative memory, or other device. As illustrated, binary CAM 302 stores a plurality of entries, wherein an entry typically contains a value and a data protection field, wherein the data protection field is a pre-computed data protection field (e.g., parity, some error correcting code, or other known or subsequently known data protection scheme) using the same data protection function as data protection generator 320.

In one embodiment, index 307 includes N bits, of which bits [1:M] correspond to the value, and bits [(M+1):N] correspond to a data protection field. The value bits of index 307 are typically used for processing corresponding to the desired application, such as retrieving a result 317 (e.g., a forwarding, routing, quality of service, or other indication) from a memory 310. The value bits of index 307 are provided to data protection generator 320, which produces a comparison value to compare mechanism 322. Compare mechanism 322 further receives the data protection bits of index 307, and makes a comparison. If these two values are equal, then compare mechanism 322 typically indicates a valid hit signal 328 and a no error signal 329. Otherwise, compare mechanism 322 typically indicates a no hit signal 328, and an error signal 329. Valid flag 316 is used to identify when signals on result 317, hit flag 328, and error flag 329 indicate an actual result.

FIG. 3B illustrates a variation of the embodiment illustrated in FIG. 3A. Binary CAM 332 (or CAM, associative memory, or device) generates an index 333 (or other value) based on input word 331. Index 333 is provided to data protection generator 340 which provides a comparison signal to compare mechanism 341. Compare mechanism 341, or any other comparison mechanism described herein, can include any mechanism for comparing two or more signals or values, such as, but not limited to one or more comparators, sets of discrete logic, processors, and/or other methods or systems for comparing two values.

Index 333 is also provided to one or more memories 334, which retrieves a pre-computed data protection field 335 provided to compare mechanism 341, and

optionally a result 347. If the comparison value pre-computed data protection field 335 are equal, then compare mechanism 341 typically indicates a valid hit signal 348 and a no error signal 349. Otherwise, compare mechanism 341 typically indicates a no hit signal 348, and an error signal 349. Valid flag 346 is used to identify when signals on result 347, hit flag 348, and error flag 349 indicate an actual result.

FIG. 3C illustrates one embodiment of a system for protecting the integrity of a ternary CAM 352 (or other CAM, associative memory, or device). In one embodiment, ternary CAM 352 maintains a set of masks and values. In some embodiments, the value bits for the masked bits, that is bits not participating in the compare, are set to zero or one. Based on an input word 351, ternary CAM 352, which may include an internal or external (not shown) priority encoder or other mechanism for selecting between multiple matches, produces an index 357. Index 357 is optionally used as input to a memory 360 or other device to produce a result 367 required for the particular application. Index 357 is also used by memory 370 (or other storage mechanism) to generate a mask 371 (corresponding to that used by the match to generate index 357) and a pre-computed data protection field 372. Input word 351 is then masked by AND device 385 using mask 371, with this result provided to data protection function 386, which produces a comparison result provided to compare mechanism 387. In one embodiment, data protection function 386 receives as input mask 371 directly, or via some other preprocessing step. In one embodiment, data protection function 386 receives the masked output of AND device 385 (and/or some other preprocessing step) as well as mask 371. Compare mechanism 387 compares this comparison value to the retrieved data protection field 372 to generate hit and error results as indicated by hit flag 388 and error flag 389. Valid flag 366 is used to identify when signals on result 367, hit flag 388, and error flag 389 indicate an actual result.

FIG. 4 illustrates another embodiment of a system for verifying the integrity of a result produced by a content-addressable memory 401 (binary, ternary or other CAM, associative memory, or device). CAM 401 produces an index 407 based on input word

400. Both input word 400 and index 407 are provided to interface 414 of data protection mechanism 410. Processor 411 manipulates the values of the received input word 400 and index 407 to determine the integrity of index 407, and to generate the integrity signals of hit flag 418 and error flag 419. Valid flag 406 is used to identify when signals on index 407, hit flag 418, and error flag 419 indicate an actual result.

In one embodiment, data protection mechanism 410 includes a processor 411, memory 412, storage devices 413, and interface 414, which are electrically coupled via one or more communications mechanisms 415 (shown as a bus for illustrative purposes). Various embodiments of data protection mechanism 410 may include more or less elements. The operation of data protection mechanism 410 is typically controlled by processor 411 using memory 412 and storage devices 413 to perform one or more tasks or processes. Memory 412 is one type of computer-readable medium, and typically comprises random access memory (RAM), read only memory (ROM), integrated circuits, and/or other memory components. Memory 412 typically stores computer-executable instructions to be executed by processor 411 and/or data which is manipulated by processor 411 for implementing functionality in accordance with the invention. Storage devices 413 are another type of computer-readable medium, and typically comprise disk drives, diskettes, networked services, tape drives, and other storage devices. Storage devices 413 typically store computer-executable instructions to be executed by processor 411 and/or data which is manipulated by processor 411 for implementing functionality in accordance with the invention. FIG. 5A-C describe several embodiments of processing performed by the embodiment illustrated in FIG. 4.

As used herein and contemplated by the invention, computer-readable medium is not limited to memory and storage devices; rather computer-readable medium is an extensible term including other storage and signaling mechanisms including interfaces and devices such as network interface cards and buffers therein, as well as any communications devices and signals received and transmitted, and other current and evolving technologies that a computerized system can interpret, receive, and/or transmit.

FIG. 5A illustrates one process for verifying the integrity of a result produced by a content-addressable or associative memory or other device. Processing begins at process block 500, and proceeds to process block 502, wherein a CAM generated index is received. Next, in process block 504, the value and data protection fields of the index are extracted. In process block 506, a data protection function is performed on the value, with this result compared in process block 508 to the extracted data protection field. If these values are equal, then in process block 510, a hit and no error conditions are indicated. Otherwise, in process block 512, a no hit and error conditions are indicated. Processing of this index is completed as indicated by process block 514.

FIG. 5B illustrates one process for verifying the integrity of a result produced by a content-addressable or associative memory or other device. Processing begins at process block 520, and proceeds to process block 522, wherein a CAM generated index is received. Next, in process block 524, a data protection field is retrieved from a storage mechanism (e.g., SRAM, memory, disk, etc.). Next, in process block 526, a data protection function is performed on the index, with this result compared in process block 528 to the retrieved data protection field. If these values are equal, then in process block 530, a hit and no error conditions are indicated. Otherwise, in process block 532, a no hit and error conditions are indicated. Processing of this index is completed as indicated by process block 534.

FIG. 5C illustrates one process for verifying the integrity of a result produced by a content-addressable or associative memory or other device. Processing begins at process block 550, and proceeds to process block 552, wherein the input word to the content-addressable memory is received. In process block 554, the CAM generated index is received. In process block 556, the mask and data protection fields corresponding to the index are retrieved, typically from a storage mechanism. Next, in process block 558, the input word is masked using the retrieved mask, with a data protection function being applied to this mask result to generate a protection value in process block 560. If, as determined in process block 562 that the protection value equals the value of the retrieved

data protection field, then in process block 564, a hit and no error conditions are indicated. Otherwise, in process block 566, a no hit and error conditions are indicated. Processing of this index is completed as indicated by process block 568.

In view of the many possible embodiments to which the principles of our invention may be applied, it will be appreciated that the embodiments and aspects thereof described herein with respect to the drawings/figures are only illustrative and should not be taken as limiting the scope of the invention. For example and as would be apparent to one skilled in the art, many of the process block operations can be re-ordered to be performed before, after, or substantially concurrent with other operations. Also, many different forms of data structures could be used in various embodiments. The invention as described herein contemplates all such embodiments as may come within the scope of the following claims and equivalents thereof.